

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Date: December 9, 2007
Robbert C. VAN DER LINDEN, et al.	Confirmation No.: 3744
Serial No.: 10/648,752	Group Art Unit: 2162
Filed: August 25, 2003	Examiner: Giovanna B. COLAN
For: METHOD AND SYSTEM FOR QUERYING STRUCTURED DOCUMENTS STORED IN THEIR NATIVE FORMAT IN A DATABASE	

Mail Stop Appeal Briefs – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

REPLY BRIEF UNDER 37 C.F.R. § 41.41

Dear Sir or Madam:

Pursuant to 37 C.F.R. § 41.41, Appellant submits this Reply Brief in response to the Examiner's
Answer mailed on August 9, 2007.

I. REAL PARTY IN INTEREST

A statement identifying the real party in interest is contained in the Appeal Brief.

II. RELATED APPEALS AND INTERFERENCES

A statement identifying the related appeals and interferences is contained in the Appeal Brief.

III. STATUS OF CLAIMS

A statement identifying the status of the claims is contained in the Appeal Brief.

IV. STATUS OF AMENDMENTS

A statement identifying the status of amendments is contained in the Appeal Brief.

V. SUMMARY OF CLAIMED SUBJECT MATTER

A summary of the claimed subject matter is contained in the Appeal Brief.

VI. GROUND OF REJECTION TO BE REVIEWED ON APPEAL

A statement identifying the grounds of rejection to be reviewed on appeal is contained in the Appeal Brief.

VII. RESPONSE TO EXAMINER'S ANSWER

Claim 1 recites:

1. A method for querying a structured document stored in its native format in a database, wherein the structured document comprises a plurality of nodes that form a hierarchical node tree, the method comprising the steps of:

(a) providing at least one child pointer within at least one of the plurality of nodes in the hierarchical node tree, wherein the at least one child pointer points to a corresponding child node in the hierarchical node tree;

(b) storing a hint within the at least one child pointer, the hint being related to the corresponding child node, wherein the at least one child pointer further comprises a node slot number of the corresponding child node; and

(c) utilizing the hint to determine whether to navigate to the corresponding child node during query evaluation.

A. Examiner's First Response

In response to Appellant's argument that Igata (U.S. Patent No. 6,853,992) does not disclose, teach, or suggest "storing a hint within the at least one child pointer, the hint being related to the corresponding child node, wherein the at least one child pointer further comprises a node slot number of the corresponding child node", as recited in claim 1, the Examiner states:

Examiner respectfully disagrees. First, the examiner cited (Fig. 12C, item 41, PART1, PART2, PART3, Col. 14, lines 28 – 33, Igata) to teach the limitation argued by the appellant above. However, the Examiner did not specifically stated that the PART1, PART2, and PART3 from the work buffer correspond to "the hint within the at least one child pointer..." claimed.

Second, Igata does disclose storing a hint within the at least one child pointer (See - Fig. 8B, "Pointer to Parent Node", "Pointer to Sibling Node", and "Pointer to Child Node", Col. 10, lines 38 – 41, "each node of the query tree has an element name, a text-

data matching condition, a pointer to a parent node, a pointer to a sibling node...", Igata; and also see- Fig. 12C, item 41, PART1, PART2, PART3, Col. 14, lines 28 – 36, Igata), the hint being related to the corresponding child node (Fig. 12B, item 41, PART1, Col. 14, lines 10 – 20, Igata). With respect to Fig. 12C, item: "PART1 0 1"; the Examiner interprets PART1 together with the arrow pointing to item: "PART2 1 2" as the child pointer claimed; wherein the PART1 corresponds to the hint claimed; and wherein, as shown in such Figure, the hint "PART1" is related to the corresponding child node ("PART2 1 2"), because "1" is included in "PART1" and is related to "PART2 1 2" ("1" in PART1 and "1" of PART2 1 2", emphasis added).

Third, Igata also discloses that: at least one child pointer further comprises a node slot number of the corresponding child node (Fig 2B, item: PART IDENTIFIER and CHILD LINK, Col. 7, lines 47 – 51, wherein the PART IDENTIFIER corresponds to the node slot number claimed, Igata).

(August 9, 2007 Examiner's Answer, pg. 15).

With respect to the Examiner's assertion that "PART1" in FIG. 12C of Igata can be construed as disclosing the "hint . . . relat[ing] to the corresponding child node" recited in claim 1 because:

"PART1" is related to the corresponding child node ("PART2 1 2"), because "1" is included in "PART1" and is related to "PART2 1 2" ("1" in PART1 and "1" of PART2 1 2", emphasis added).

(August 9, 2007 Examiner's Answer, pg. 15), Appellant respectfully directs the Examiner to the following passage of Igata:

As shown in FIG. 2B, each node has an element name, an offset in a row of sibling parts in the document, a part-ID for identifying a meta part, a link to a parent node (a parent link), a link to a sibling node (a sibling link), and a link to a child node (a child link)

(col. 7, lns. 47-51 of Igata).

As clearly seen from the passage above and from FIGs. 2A-2B and 12B-12C of Igata, for the node "PART1 0 1", "PART1" is the element name of the document part corresponding to the node, "0" is the offset in the document for the document part corresponding to the node, and "1" is the part-ID for the document part corresponding to the node. For the node "PART2 1 2", "PART2" is the element name of the document part corresponding to the node, "1" is the offset in the document for the document part corresponding to the node, and "2" is the part-ID for the document part corresponding to the node.

Thus, the “1” in node “PART1 0 1” has absolutely nothing to do with the “1” in node “PART2 1 2”. In fact, as clearly shown in FIGs. 2A and 12B-12C of Igata, node “PART2 1 2” is not even a child of node “PART1 0 1”. Rather, node “PART1 0 1” and node “PART2 1 2” are sibling nodes that correspond different parts of the same document, i.e., “<PART1>” and “<PART2>” in “Document 1”, which is clearly set forth in column 7, lines 30-40, and column 13, lines 40-52 of Igata. Therefore, the “PART1” in FIG. 12C of Igata cannot be construed as disclosing, teaching, or suggesting the “hint . . . relat[ing] to the corresponding child node”, as recited in claim 1.

Additionally, claim 1 recites “the structured document comprises a plurality of nodes that form a hierarchical node tree”. Thus, the “hierarchical node tree” recited in claim 1 is a hierarchical node tree of a “structured document”.

In contrast, FIG. 8 of Igata cited by the Examiner “is a diagram showing an example structure of a query tree” (col. 6, Ins. 1-2 of Igata) (emphasis added). The “query tree” in Igata is created “from a user’s query” (col. 10, ln. 12 of Igata). Since Igata shows a “hierarchical index” tree in FIG. 2 created from a “Document 1” and a “Document 2” that is separate and distinct from the “query tree” in FIG. 8, the “query tree” in Igata cannot be construed as disclosing, teaching, or suggesting the “hierarchical node tree” recited in claim 1.

With regards to the “work buffer 41” in Igata relied upon by the Examiner, Appellant respectfully directs the Examiner to the following passage of Igata:

FIG. 5 is a flowchart showing the steps of processing for registration in the hierarchical index 13 (details of the processing in step S18 of FIG. 3). This processing includes the following four stages.

- 1) Following the tree structure up to a node corresponding to the upper meta part (step S31);
- 2) When the upper meta part has no child (step S32), extending a child link of the upper meta part to the new node (step S33), and when the upper meta part has a child, adding the new node (sibling link) to the end of the child node of the upper meta part (step S34);

- 3) Extending a link (parent link) from the new node to the upper meta part (step S35); and
- 4) Extending an element link to a node having the same element name.

The processing of extending an element link described in the above stage 4) is performed by means of the following steps.

- (a) Preparing a work buffer in which element names are registered and from which links are extended to corresponding nodes;
- (b) When the work buffer is empty (no element name is registered) (step S36), registering a new node in the work buffer (step S37); and
- (c) When a node has been registered in the work buffer (step S36), extending an element link from the registered node to the new node, and registering the new node in the work buffer (step S38).

(Col. 8, ln. 55 to col. 9, ln. 15 of Igata).

As evident from the passage above, the “work buffer” in Igata is used to establish “element links” between nodes having the same element name and is completely unrelated to parent-child relationships between nodes. In fact, the “element link” is actually used to establish relationships between parts of different documents. For instance, in the “hierarchical index” tree shown in FIG. 2A of Igata, there is an element link from node “PART1 0 1”, which corresponds to a part in “Document 1”, to node “PART1 1 5”, which corresponds to a part in “Document 2”. There is also an element link from node “PART2 1 2”, which corresponds to a part in “Document 1”, to node “PART2 0 4”, which corresponds to a part in “Document 2”.

With respect to the Examiner’s assertion that the “Part Identifier” in each node of “hierarchical index” tree shown in FIGs. 2A-2B of Igata corresponds to the “node slot number of the corresponding child node” recited in claim 1, Appellant respectfully directs the Examiner to read the passage of Igata cited by the Examiner, which states:

As shown in FIG. 2B, each node has an element name, an offset in a row of sibling parts in the document, a part-ID for identifying a meta part, a link to a parent node (a parent link), a link to a sibling node (a sibling link), and a link to a child node (a child link).

(col. 7, lns. 47-51 of Igata).

From the cited passage, it is clear that the “part-ID” in each node of “hierarchical index” tree in Igata is an identifier for the document part corresponding to the node and not an identifier for a child of the node. In particular, looking at the “hierarchical index” tree shown in FIG. 2A of Igata, the node “DOCUMENT 0 0” with a part-ID of “0” has a child link to node “PART1 0 1”. The part-ID “0” in node “DOCUMENT 0 0” is completely unrelated to child node “PART1 0 1”. Hence, contrary to the Examiner’s assertions, the “part-ID” in each node of “hierarchical index” tree in Igata cannot be construed as disclosing, teaching, or suggesting the “node slot number of the corresponding child node” recited in claim 1.

In addition, Appellant respectfully requests the Examiner to clarify which element in Igata the Examiner is construing as disclosing the “child pointer” recited in claim 1. Specifically, the Examiner first cites the “POINTER TO CHILD NODE” in FIG. 8 of Igata as disclosing the “child pointer” recited in claim 1. However, the Examiner then cites the “CHILD LINK” in FIG. 2 of Igata as disclosing the “child pointer” recited in claim 1. Appellant respectfully reminds the Examiner that FIG. 8 of Igata is directed to a “query tree”, whereas FIG. 2 of Igata is directed to a “hierarchical index” tree. Thus, Appellant respectfully submits that the Examiner cannot simply substitute one for the other whenever it is convenient for the Examiner to do so.

As seen from above, assertions made by the Examiner are, at best, unintentional misinterpretations of Igata and, at worst, deliberate misrepresentations of what is taught by Igata. Specifically, the Examiner not only cites irrelevant material from Igata (e.g., the “query tree” and the “work buffer” in Igata), the Examiner also makes broad statements that are completely contrary to what is taught in Igata (e.g., asserting that the “1” in node “PART1 0 1” is a hint to the “1” in node “PART2 1 2”; that the link from node “PART1 0 1” to “PART2 1 2” is a child link; and that the “part-ID” of a node is

an identifier for a child node) and arbitrarily combines different aspects of Igata (e.g., mixing aspects of the “query tree” with aspects of the “hierarchical index” tree).

Appellant does not dispute that Igata discloses a “hierarchical index” tree created from documents. Nor does Appellant dispute that a node in the “hierarchical index” tree of Igata may include a “child link” to a child node. This is clearly shown in FIGs. 2A-2B and discussed from column 7, line 15 to column 9, line 15 of Igata.

What Appellant is disputing, and has been disputing for some time, is that the “child link” in Igata cannot be construed as disclosing, teaching, or suggesting the “child pointer” recited in claim 1 because (a) the “child link” in Igata does not have stored therein a “hint . . . relat[ing] to the corresponding child node”, as recited in claim 1, and (b) the “child link” in Igata does not comprise a “node slot number of the corresponding child node”.

As discussed above, Igata does not disclose, teach, or suggest a “hint . . . relat[ing] to the corresponding child node” or a “node slot number of the corresponding child node”. In addition, even if it is assumed argumentatively that “PART1” in Igata can be construed as disclosing a “hint”, as asserted by the Examiner, and the “part-ID” can be construed as disclosing a “node slot number”, as asserted by the Examiner, Igata does not disclose, teach, or suggest that “PART1” and “part-ID” are contained inside the “child link”.

In particular, as illustrated in Figure 8 and described on pages 13-14 of the present application, the “child pointer” recited in claim 1 is more than just a link to another node. The “child pointer” in the present application is defined in such a way that it includes a “hint . . . relat[ing] to the corresponding child node” and a “node slot number of the corresponding child node”. Igata does not disclose, teach, or suggest that the “child link” is anything more than just a link to a child node.

Further, it is not inherent that the “child link” in Igata is more than just a link to a child node.

Under M.P.E.P. § 2163:

To establish inherency, the extrinsic evidence “must make clear that the missing descriptive matter is necessarily present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill. Inherency, however, may not be established by probabilities or possibilities. The mere fact that a certain thing may result from a given set of circumstances is not sufficient.” *In re Robertson*, 169 F.3d 743, 745, 49 U.S.P.Q.2d 1949, 1950-51 (Fed. Cir. 1999).

(M.P.E.P. § 2163.07, 8th ed., Sept. 2007 rev.). Since persons of ordinary skill in the art will readily understand that a link to a child node can simply be a link and not include anything else, the Examiner cannot argue that Igata inherently discloses such.

Accordingly, Igata does not disclose, teach, or suggest “storing a hint within the at least one child pointer, the hint being related to the corresponding child node, wherein the at least one child pointer further comprises a node slot number of the corresponding child node”, as recited in claim 1. Therefore, claim 1, and the claims that depend therefrom, are not anticipated by Igata. Given that claims 11 and 21 each recite elements similar to those of claim 1, claims 11 and 21, and the claims that depend therefrom, are not anticipated by Igata for at least the same reasons.

B. Examiner’s Second Response

In response to Appellant’s argument that Igata does not disclose, teach, or suggest “utilizing the hint to determine whether to navigate to the corresponding child node during query evaluation”, as recited in claim 1, the Examiner states:

Examiner respectfully disagrees. Igata does disclose utilizing the hint to determine whether to navigate to the corresponding child node during query evaluation (Col. 9 and 21, lines 29 – 33 and 21 – 27; respectively, “In the present embodiment, a set (special search key) which includes a special character string and a part-ID is registered for each document part, which enables searching of a structure only (search which is performed under a single search condition that a certain document part is present)...”, and “...As described above, in the present invention, there are provided a hierarchical index which expresses the structure of each structured document, and a text index

which is used for searching a document-ID on the basis of a search key composed of a part-ID and a character string in text data; a user's query received in the form of a tree structure is converted to a Boolean expression; and the text index is referred to in order to obtain a document-ID corresponding to the query tree. Thus, it becomes possible to search structured documents, while maintaining the high speed of conventional full-text search engines..."; wherein the step of searching including a search key corresponds to the step of navigating as claimed; wherein the step of searching under the single condition that a certain document part is present corresponds to navigating to the child node; wherein the part corresponds to the child node claimed, Igata).

(August 9, 2007 Examiner's Answer, pg. 16).

First, Appellant respectfully reminds the Examiner that the Examiner asserts on page 15 of the Examiner's answer that the element name "PART1" in Igata discloses the "hint" recited in claim 1 and that the "part-ID" in Igata discloses the "node slot number" recited in claim 1. Therefore, the Examiner cannot now assert that the "part-ID" or the "document-ID" in Igata discloses the "hint" recited in claim 1.

Second, as previously discussed, neither the "element name" nor the "part-ID" in a node of the "hierarchical index" tree in Igata are in any way related to a child of the node. Hence, neither the "element name" nor the "part-ID" in Igata can be construed as disclosing the "hint . . . relat[ing] to the corresponding child node" recited in claim 1.

Third, Appellant respectfully requests the Examiner to focus on what is actually recited in claim 1. Specifically, claim 1 recites "utilizing the hint to determine whether to navigate to the corresponding child node during query evaluation". In other words, a "hint" stored within a "child pointer" provided within a "node" of a "hierarchical node tree" representing a "structured document" is used to determine whether to navigate from the "node" of the "hierarchical node tree" to a "child node" of the "node" in the "hierarchical node tree".

Claim 1 does not recite using a “text index” or a “query tree” to determine whether to navigate from one node of the “hierarchical node tree” to another node of the “hierarchical node tree”. Thus, it is irrelevant that lgata teaches that “a user’s query [is] received in the form of a tree structure” and “a text index . . . is used for searching a document-ID on the basis of a search key composed of a part-ID and a character string text data”.

Accordingly, lgata does not disclose, teach, or suggest “utilizing the hint to determine whether to navigate to the corresponding child node during query evaluation”, as recited in claim 1. Therefore, claim 1, and the claims that depend therefrom, are not anticipated by lgata. Given that claims 11 and 21 each recite elements similar to those of claim 1, claims 11 and 21, and the claims that depend therefrom, are not anticipated by lgata for at least the same reasons.

Claim 31 recites:

31. The method of claim 1, wherein each of the plurality of nodes in the hierarchical node tree specifies a type of node, one or more nodes in the hierarchical node tree being of a text type and one or more other nodes in the hierarchical tree being of a non-text type.

C. Examiner’s Third Response

In response to Appellant’s argument that lgata does not disclose, teach, or suggest “wherein each of the plurality of nodes in the hierarchical node tree specifies a type of node, one or more nodes in the hierarchical node tree being of a text type and one or more other nodes in the hierarchical tree being of a non-text type”, as recited in claim 31, the Examiner states:

Examiner respectfully disagrees. lgata does disclose the limitation where each of the plurality of nodes in the hierarchical node tree specifies a type of node (Col. 11 – 12, lines 24 – 28 and 7 – 9, a node type; respectively, lgata), one or more nodes in the hierarchical node tree being of a text-type (Col. 4 and 11, lines 24 – 31, and 25 – 28; respectively, text data of each document, lgata) and one or more other nodes in the hierarchical tree being of non-text type (Col. 4, lines 24 – 31, meta – part, lgata). Wherein the tree structure (Col. 4, lines 24 – 26, lgata) corresponds to the hierarchical node tree as claimed. To add, the tree structure of lgata includes both text nodes and

non-text nodes (Col. 4, lines 26 – 29, in a tree structure in which a "meta part" ... ; a text index in which a character string contained in text data of each document is registered, Igata). And to further clarify, as disclosed by Igata, in Fig. 9, items S61, S62, the query tree and structure of hierarchical index are attach nodes and assemble nodes to create Boolean logic tree. Such citations teaches that the structures are attached as shown by Igata, in Fig. 18B, where as shown one or more nodes being of a text type (for example, node including "Document 0 0") and non-text type (for example, node including "? ? 2").

(August 9, 2007 Examiner's Answer, pg. 17).

The first passage cited by the Examiner states:

Each temporary node serves as a node of a Boolean logic tree. Such a temporary node is created to have a node type (AND, OR, the type of a search key), a text-data matching condition, and a part-ID. A node whose type is AND or OR is used as an intermediate node, and ultimately, a single logic tree is assembled in step S62 of FIG. 9.

(Col. 11, Ins. 23-28 of Igata).

As seen from above, the first cited passage of Igata is clearly discussing the assembly of a "Boolean logic tree" for a query, which cannot be construed as disclosing the "hierarchical node tree" recited in claim 31 because Igata discloses a "hierarchical index" tree that is separate and distinct from the "Boolean logic tree". In addition, the "node type" discussed in the cited passage of Igata is one of "AND", "OR", or "type of a search key". There is nothing about the "node type" being "text" or "non-text".

The second passage cited by the Examiner states:

(e) When the type of a child of the parent node of the query tree does not match the type of a child of an AND node, discarding the AND node (step S88), and attaching to the node obtained through the processing in step S82 an AND node which was not discarded in step S88) (step S89).

(Col. 12, Ins. 7-12 of Igata).

From the second cited passage, it is clear that Igata is discussing the "query tree", which, as previously discussed, cannot be construed as disclosing the "hierarchical node tree" recited in claim 31

because the “query tree” is created from a user query, whereas the “hierarchical node tree” is created from a “structured document”.

The third passage cited by the Examiner states:

The present invention solves the above-described problems involved in the conventional methods through provision of a hierarchical index which expresses the structure of each structured document such that the hierarchical relationship among document parts is expressed in a tree structure in which a "meta part" is treated as a single node; a text index in which a character string contained in text data of each "document part" is registered; and search means which receives a user's query in a tree structure and converts it to a Boolean expression. The term "document part" refers to each element part of each structured document, and the term "meta part" refers to each element part which is common among several structured documents.

(Col. 4, Ins. 22-35 of Igata).

Even though the third passage of Igata cited by the Examiner explicitly states that the “hierarchical index” tree only includes the “meta part” of a “structured document” and that the “text data” is stored in a separate “text index”, the Examiner still insists that the node “Document 0 0” in “hierarchical index” tree is a node of “text type”.

Moreover, the temporary node “? ? 2” in FIG. 18B of Igata cited by the Examiner is a temporary node that is to be used to form the “Boolean logic tree” as shown in FIGs. 21B-21C of Igata. The temporary node is not a node of the “hierarchical index” tree in Igata, which the Examiner cites as disclosing the “hierarchical node tree” recited in claim 31.

Accordingly, Igata does not disclose, teach, or suggest “wherein each of the plurality of nodes in the hierarchical node tree specifies a type of node, one or more nodes in the hierarchical node tree being of a text type and one or more other nodes in the hierarchical tree being of a non-text type”, as recited in claim 31. Therefore, claim 31 is not anticipated by Igata. Given that claims 32 and 33 each

recite elements similar to those of claim 31, claims 32 and 33 are not anticipated by Igata for at least the same reasons.

Claim 34 recites:

34. The method of claim 1, wherein the at least one node in the hierarchical node tree further includes at least one other child pointer, the at least one other child pointer pointing to itself or to an in-lined character array.

D. Examiner's Fourth Response

In response to Appellant's argument that Igata does not disclose, teach, or suggest "wherein the at least one node in the hierarchical node tree further includes at least one other child pointer, the at least one other child pointer pointing to itself or to an in-lined character array", as recited in claim 34, the Examiner states:

Examiner respectfully disagrees. The applied reference does disclose one node in the hierarchical node tree further includes at least one other child pointer (Fig. 12C, item: node PART1, Col. 14, lines 28 – 33, Igata), the at least one other child pointer pointing to itself or to a in-lined character array (Fig. 12C, item: node PART1, PART2, PART3, and PART2, Col. 14, lines 28 – 33, Igata). Wherein the pointer of the node including PART1 corresponds to the child pointer claimed; and the nodes including PART2, PART3, and PART2 correspond to the in-lined character array claimed. To further clarify, the Examiner interprets the nodes including PART2, PART3, and PART2 as the in-lined character array because such nodes are in lined linked to each other and comprise characters. Additionally, Examiner makes note that the limitation "...pointer pointing to itself or to an in-lined character array..." includes the optional limitation "or". Therefore, the limitation recites whether "pointer pointing to itself" or "to an in-lined character array".

(August 9, 2007 Examiner's Answer, pg. 18).

Appellant respectfully reminds the Examiner that although the Examiner is entitled to a reasonably broad interpretation of claim terms, the Examiner cannot select an interpretation that is inconsistent with the specification or contrary to the meaning of a term accepted by those of ordinary skill in the art. M.P.E.P. § 2111 states:

[The] “broadest reasonable interpretation [must be] consistent with the specification.” The Federal Circuit’s en banc decision in *Phillips v. AWH Corp.*, 415 F.3d 1303, 75 USPQ2d 1321 (Fed. Cir. 2005) The broadest reasonable interpretation of the claims must also be consistent with the interpretation that those skilled in the art would reach. *In re Cortright*, 165 F.3d 1353, 1359, 49 USPQ2d 1464, 1468 (Fed. Cir. 1999).

(M.P.E.P. § 2111, 8th ed., Sept. 2007 rev.).

The Examiner asserts that “nodes . . . PART2, PART3, and PART2” in FIG. 12C of Igata can be construed as disclosing the “in-lined character array” recited in claim 34 because “such nodes are in lined linked to each other and comprise characters”. Claim 34 and the specification of the present application, however, distinguishes between nodes and arrays.

In addition, persons of ordinary skill in the art would not construe a bunch of nodes as an “in-lined character array” simply because the nodes are linked together in a line. Further, page 8, lines 14-15 of the present application specifically states that an “in-lined character array . . . contains information describing the child [node]”. As previously discussed, nodes “PART2”, “PART3”, and “PART2” in Igata are not child nodes of node “PART1”. In fact, based on the examples in Igata, node “PART1” does not have a child node at all. Thus, “nodes . . . PART2, PART3, and PART2” in FIG. 12C of Igata cannot be construed as disclosing, teaching, or suggesting the “in-lined character array” recited in claim 34.

Accordingly, Igata does not disclose, teach, or suggest “wherein the at least one node in the hierarchical node tree further includes at least one other child pointer, the at least one other child pointer pointing to itself or to an in-lined character array”, as recited in claim 34. Therefore, claim 34 is not anticipated by Igata. Given that claims 35 and 36 each recite elements similar to those of claim 34, claims 35 and 36 are not anticipated by Igata for at least the same reasons.

CONCLUSION

On the basis of the above remarks, and the remarks made in the Appeal Brief, Appellant respectfully submits that the final rejection should be reversed.

Respectfully submitted,
SAWYER LAW GROUP LLP

Dated: December 9, 2007

/Erin C. Ming/
Erin C. Ming
Attorney for Applicant
Reg. No. 47,797
(650) 475-1449

VIII. APPENDIX OF CLAIMS

A listing of the claims involved on appeal is contained in the Appeal Brief.

IX. EVIDENCE APPENDIX

None

X. RELATED PROCEEDINGS APPENDIX

None